Event-driven simulation of neural population synchronization facilitated by electrical coupling

Journal:	BioSystems
Year of Publication:	2007
Volume:	87
Issue:	2-3
Pages:	275-280
DOI	10.1016/j.biosystems.2006.09.023
Manuscript Type:	Regular paper
Date Submitted by the Author:	28-Feb-2005
Complete List of Authors:	Richard R. Carrillo; Department of Computer Architecture and Technology, E.T.S.I. Informática, University of Granada Eduardo Ros; Department of Computer Architecture and Technology, E.T.S.I. Informática, University of Granada Boris Barbour; Laboratoire de Neurobiologie (CNRS UMR 8544), Ecole Normale Supérieure Christian Boucheny; Evasion - Laboratoire GRAVIR INRIA Rhône-Alpes ; Laboratoire de Physiologie de la Perception et de l'Action, Collège de France Olivier Coenen; Sony Computer Science Laboratory Paris
Keywords:	Event-driven, Spiking neuron, Neural synchronization, Electrical coupling

Event-driven simulation of neural population synchronization facilitated by electrical coupling

Richard R. Carrillo, Eduardo Ros, Boris Barbour, Christian Boucheny, Olivier Coenen

Abstract

Most neural communication and processing tasks are driven by spikes. This has enabled the application of the event-driven simulation schemes. However the simulation of spiking neural networks based on complex models that cannot be simplified to analytical expressions (requiring numerical calculation) is very time consuming. Here we describe briefly an event-driven simulation scheme that uses pre-calculated table-based neuron characterizations to avoid numerical calculations during a network simulation, allowing the simulation of large-scale neural systems. More concretely we explain how electrical coupling can be simulated efficiently within this computation scheme, reproducing synchronization processes observed in detailed simulations of neural populations.

Keywords: Event-driven; Spiking neuron; Neural synchronization; Electrical coupling

Introduction

One of the abilities of biological neural systems is their parallel information processing. The study of the dynamics of the cells that form these massively parallel computation systems is still an open issue (D'Angelo et al., 2001; Koch, 1999). Most of the computations that take place in these systems are spike driven: a spike that arrives to a target cell affects its state, producing a transient behaviour. Besides, in some models, the neural state evolution can be predicted. This has motivated the development of event-driven simulation schemes (Delorme and Thorpe, 2003; Mattia and Guidice, 2000; Reutimann et al., 2003). Some approaches simulate simple neural models in which the new neural state can be calculated after an input spike with a simple expression (Delorme and Thorpe, 2003). Other approaches use iterative calculation during the simulation to obtain the future neuron state of more complex models (Makino, 2003). Some authors use lookup tables to support concrete features such as stochastic dynamics (Reutimann et al., 2003). In our approach, the complete neural dynamics are compiled into lookup tables to enable the efficient simulation of detailed neural models which traditionally required time-driven approaches. Therefore during the event-driven simulation, only table accesses are needed in order to calculate the neuron state evolution.

The short-term dynamics defined by complex differential equations which require numerical calculation are computationally costly to simulate. We have developed an event-driven computation scheme that uses pre-calculated short-term dynamics which are stored on cell characterization tables, enabling the simulation of models of different degree of complexity (limited by the size of the tables required to store the model). In this way, an event-driven computation scheme in which the cells states are only updated at the arrival of spikes represents a very efficient tool for the simulation of large-scale systems. Network long-term dynamics (for instance learning models) can be simulated on a different time scale. We can adopt an event-driven scheme if the leaning mechanisms to be simulated are driven by spikes.

The two assumptions that are done to develop the event-driven computation scheme based on tables are the following: (a) the effect of a spike on a target neuron state can be predicted (since the simulation time jumps from one event to another, the neuron state must be also updated discontinuously, as indicated in Section 2) and (b) the number of inter-related variables that define the cell model dynamics is not very large (this makes the number of needed table dimensions affordable).

The process of building up cell models and setting up a system scale simulation requires of different stages:

(1) Detailed neuron model simulation.
Detailed simulations of neural models are done with specific tools such as NEURON (Hines and Carnevale, 1997) or GENESIS (Bower and Beeman, 1998). A simplification process leads to simplified models characterized by a reduced number of variables and differential equations.

(2) Table definition.
We define the table structure that will be used by the simulator to calculate the neural state evolution of a synaptic-conductance-based neural model online quickly, for instance synaptic conductance decay $g_{\mathrm{exc}}(t)$ and $g_{\mathrm{inh}}(t)$, firing time prediction $t_{\mathrm{f}}(V_{\mathrm{m},t_0},\ g_{\mathrm{exc},t_0},\ g_{\mathrm{inh},t_0})$ and the membrane potential evolution $V_{\mathrm{m}}(V_{\mathrm{m},t_0},\ g_{\mathrm{exc},t_0},$

g_{inh,t_0}, Δt). All these variables depend on their previous states (at the last time they where updated t_0), the previous state of other variables or the time elapsed t since then.

(3) Compilation of the characterization tables.
All the transient dynamics of the cells are simulated off-line. This requires massive numerical computation to sample the cell behaviour accurately when using complex neural models. This massive computation consists roughly on single cell simulations under different conditions. This can be easily parallelized in clusters of computers if a very fast table creation process is required.

(4) Efficient neural system simulation.
We run the event-driven simulation scheme that uses efficiently these tables to avoid online numerical calculations (this is briefly described in the next section).

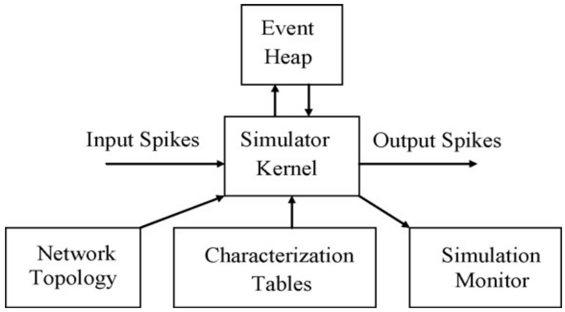

Fig. 1. Block diagram of components of the event-driven simulation scheme.

In the next section we describe briefly the table-based event-driven computation scheme (see Fig. 1). In Section 3 we describe how electrical synapses can be simulated in the presented computational approach. Furthermore, we show some illustrative results of neural population synchronization processes facilitated by electrical coupling. This experiment is motivated by different reasons: (1) validation of the implementation of electrical synapses, (2) the event-driven simulation of neural synchronization processes in highly interconnected large networks with arbitrary delays may become a challenge, because a large number of events are fired in a short time interval (which may lead to saturation of the event reordering data structure), and finally (3) synchronization processes seem to play an important role in the computations occurring within the molecular layer of the cerebellum, and should be integrated in further cerebellar simulations (we plan to study the role of this computational primitive in the sparse coding that is generally assumed to take place within the cerebellar granular layer).

1. Table-based event-driven computation scheme

The characterization of the neural dynamics requires a finite number of cell simulations under different initial conditions during a finite time interval (e.g. 50 ms).

Therefore, we sample the neural behaviour in a number of possible transient dynamics. A priori knowledge about the waveform of the target functions helps to optimize the table's size by compression techniques such as logarithm sampling of exponential-like functions.

In the event-driven scheme the neuron variables need to be updated only when the cell receives or fires a spike (since the firing time is delayed and arbitrary delays are allowed, two events are needed to be generated for each neuron spike). Therefore, the simulation time (t) jumps from one event to next one. It is mandatory to process all the events of the network in chronological order. This requires a re-ordering process each time that a new event is produced or processed. In order to optimize the time required for this task we have chosen a "heap data structure" (Aho et al., 1974; Williams, 1964) to store the input and output events. This data structure minimizes the time required by the re-ordering process, offering a good performance even for a high global activity or heap occupancy. Each time that an output spike is produced, a connection table is consulted and input spikes to the target neurons are inserted into the event heap sequentially (in order not to overload this data structure) according to the connection delays. Events affecting a cell state may result in the invalidation of some predicted spikes of the heap. This is checked out each time that a new event is processed.

The computation scheme processes two kinds of events:

- Output event.
  - Update the neural variables to the post-firing state.

- Insert a new output event into the heap if the neuron is able to fire again in the absence of stimuli.
  - Insert the input event with the shortest delay of the connection table into the spike heap.
- Input event.
  - Update the neural variables consulting the characterization tables.
  - Insert a new output event into the heap if the neuron is able to fire in the absence of further stimuli.
  - Insert the next input event from the connection table (the next one with the shortest delay).

The implemented synaptic-conductance-based neural model with delayed firing (Ros et al., 2006) can be mapped into the following characterization tables (A more complex neural model would require a greater number of tables):

- *Synaptic conductances*: $g_{\mathrm{exc}}(t)$ and $g_{\mathrm{inh}}(t)$ are tables used to update the conductance values depending on the time elapsed since the last input spike.
- *Firing time*: $t_{\mathrm{f}}(V_{\mathrm{m},t_0}, g_{\mathrm{exc},t_0}, g_{\mathrm{inh},t_0})$ is a table used to predict the time of next output spike produced by the cell if it does not receive any further stimuli.
- *Membrane potential*: $V_{\mathrm{m}}(V_{\mathrm{m},t_0}, g_{\mathrm{exc},t_0}, g_{\mathrm{inh},t_0}, \Delta t)$ is a table describing the membrane potential evolution after receiving an input spike.

The event-driven computation scheme allows the simulation of large-scale spiking networks. The computation speed depends on the network activity (spikes per second) almost linearly whereas the network size has little influence. With a conventional computing platform (Pentium IV at 1.8 GHz) we are able to process 8×10^5 spikes/s. We have evaluated the performance of the computation scheme with different network sizes and average activity. For instance, 1 s of simulation of a network of 8×10^4 cells with an average firing rate of 10 Hz takes less than a second, thus it can be done in real time.
After characterizing different types of cerebellar neurons (Granule, Golgi, Purkinje, deep cerebellar nuclei cells and interneurons), the described approach is being used to simulate in real-time cerebellar adaptive models (Boucheny et al., 2005). Currently, simulations on a dual Pentium IV 2.8 GHz platform, of a cerebellar model of 2080 cells with 52,000 synaptic connections and a global activity of approximately 10^6 spikes per second, runs in real-time including learning and input/output translations related with robot control (Boucheny et al., 2005).

2. Neural population synchronization facilitated by electrical coupling

It is believed that electrical synapses facilitate the synchronous firing of interconnected cells (Chez, 1991; Kopell and Ermentrout, 2004; Kepler et al., 1990; Traub and Bibbig, 2000; Draghun et al., 1998). These synapses are characterized by extremely fast transients, through direct flow current. The gap junctions usually have a very low conductance (approximately 100 pS according to Neyton and Trautmann, 1985), so we neglect subthreshold electrical coupling. This assumption directly allows the efficient simulation of electrical synapses on an event-driven scheme. In this way, a neuron only affects other cells connected by electrical synapses when an action potential is fired. During the action potential effect (1.5 ms approximately) we increase the membrane potentials of the connected cells by an amount that depends on the coupling ratio (electrical connection weight). Unidirectional electrical synapses have been documented (Furshpan and Potter, 1959), therefore we implement internally unidirectional coupling since bidirectional coupling can be simulated defining two unidirectional connections.

1.1. Implementation of electrical connections on an event-driven scheme

In one possible implementation, when a neuron with electrical synapses fires a spike, two events are inserted into the heap:

- *Starting event*. Indicating the initial time of electrical coupling effect. In fact, normally no delay is introduced (although it is allowed by the simulation scheme) since this kind of synapses is characterized by its rapid response. When this event is processed the simulation kernel increments the membrane potential of the target cell by an amount that depends on the connection weight.
- *Ending event*. Indicating the termination of the electrical coupling on the target neurons. When this event is processed, the simulation kernel decrements the membrane potential of the target neuron in the same amount that it was increased by the staring event.

Usually an interval of 1.5 ms is left between the starting and ending events. In this way, the effect of electrical coupling is a very fast increment of the membrane potential of the target neurons during a short time interval. As commented before, the electrical coupling is driven by action potentials since we are neglecting sub-threshold electrical coupling.

This implementation has been discarded because the large amount of generated ending events that need to be stored on the event reordering structure when the starting event is processed, producing a computational bottleneck.

Another choice that has been tested is the inclusion of a single event that initiates a triangular spikelet on the target neuron membrane potential. In order to implement this, the neuron includes a variable that stores the instant at which the effect finishes and the current amplitude of the spikelet (defined by the strength of the coupling). When the membrane potential is updated due to other events, these variables are consulted to know if there is any spikelet still present in the neuron membrane potential and to calculate its current amplitude (the amplitude of the simulated spikelet decrements linearly. See Fig. 2). The final membrane potential is calculated adding its current value and the current spikelet amplitude.

1.2. Simulation of neural population synchronization processes

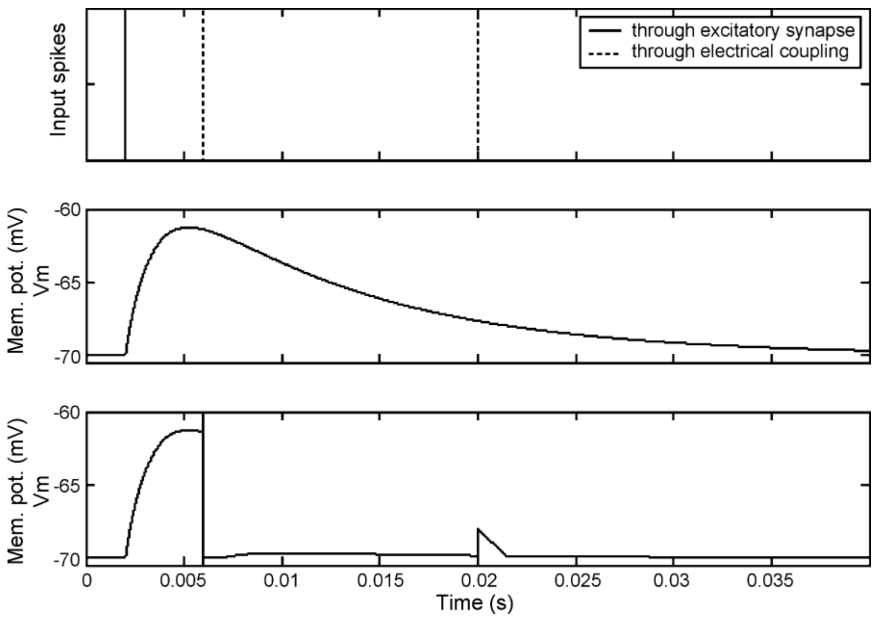

Fig. 2. Illustration of the effect produced by electrical coupling in the simulation. The upper plot show the input spikes. The middle plot illustrates the membrane potential evolution in the absence of electrical coupling. The bottom plot illustrates the spikelets (coupling potentials) produced by the electrical coupling. In fact, since the membrane potential of the cell is closed to the firing threshold when it receives the first spike through the electrical connection, it forces the neuron to fire synchronously.

Electrical coupling has been proven to be an effective synchronization mechanism (Kopell and Ermentrout, 2004; Kepler et al., 1990; Traub and Bibbig, 2000; Draghun et al., 1998) and there are many examples of electrical coupling between inhibitory neurons in the nervous system (Gibson et al., 1999; Long et al., 2004; Mann-Metzer and Yarom, 1999). Here we want to evaluate the simulation of electrical coupling within an event-driven scheme. For this purpose, we simulate a neural network of 100 cells receiving spikes at an average rate of 200 Hz with a standard deviation of 0.1 through excitatory synapses. These input spikes encode a constant bias and a random component. The cells are interconnected with inhibitory synapses and electrical coupling with an all-to-all topology. The network consists of 100 neurons with 100 input excitatory synapses (one per cell), 10,000 inhibitory synapses and 10,000 electrical connections. We have used neurons that intend to emulate cerebellar interneurons (Ros et al., 2006), using the following characterization parameters: membrane capacitance $C_m = 30$ pF; time constants of the excitatory and inhibitory synapses $\tau_{exc} = 0.5$ ms and $\tau_{inh} = 2$ ms; resting conductance $G_{rest} =$ 0.2 nS; excitatory and inhibitory reversal potentials $E_{exc} = 0$ V and $E_{inh} = -80$ mV; resting potential $E_{rest} = -70$ mV; firing threshold $V_{th} = -60$ mV. This cell profile has been used to extract the characterization tables through intense numerical calculation using a conductance-based-synaptic-input neural model (Gerstner and Kistler, 2002) before the event-driven simulation. The computing scheme processes everything in real-time (i.e. the computation time is much shorter than the simulated time; 1 s of simulation takes about 0.4 s), since no numerical calculation is required during the event-driven simulation.

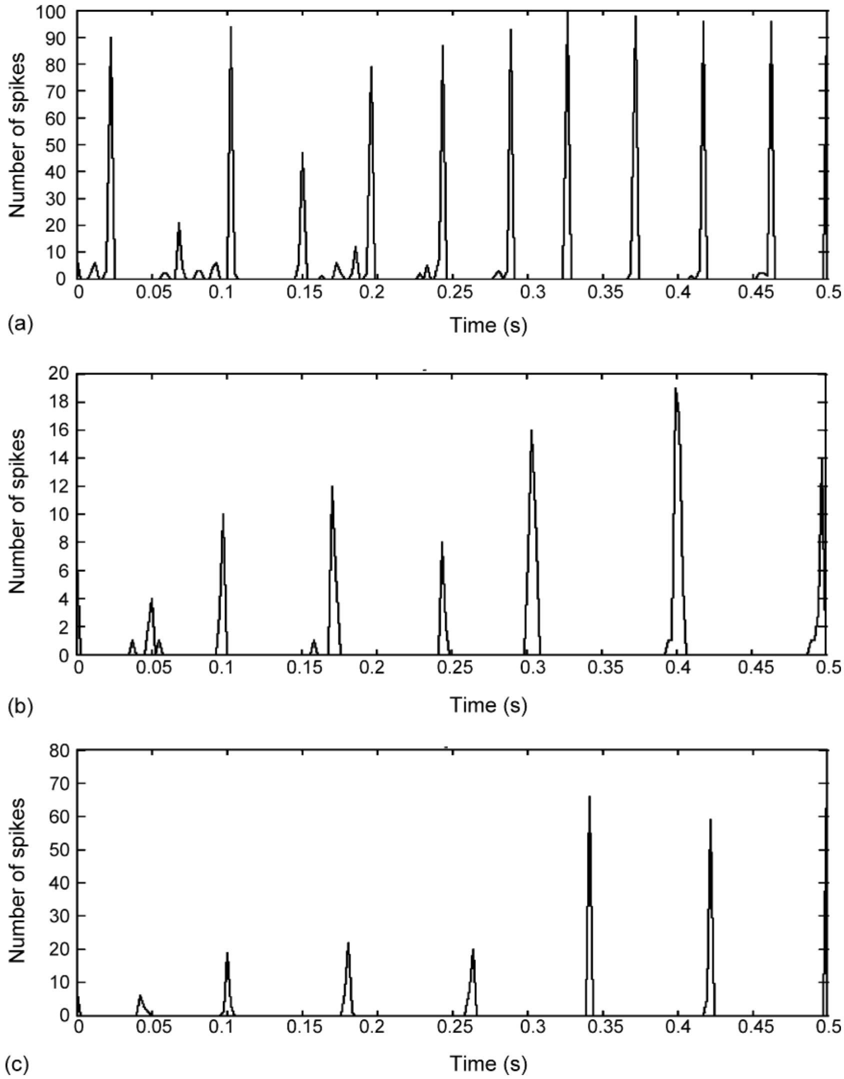

Fig. 3. Neural population synchronization histograms. (a) Only electrical coupling with coefficient 0.02. (b) Only inhibitory synapses with G_{inh} = 1.65 nS. (c) Inhibitory synapses (G_{inh} = 1.65 nS) and electrical coupling (coefficient of 0.02); there are no neurons firing asynchronously almost since the beginning (the frequency is higher in (a) because there is no inhibition).

In Fig. 3 we show the obtained synchronization histograms using inhibition and electrical coupling. These results are very similar to the ones obtained in Kopell and Ermentrout (2004) using a network of quadratic integrate-and-fire neurons (Latham et al., 2000); the synchronization was created quickly and multiple clusters of cells were not observed (see Fig. 3). This validates our electrical coupling approach and proves event-driven simulation scheme to be an efficient tool to study this kind of processes or to apply them in neural network running in real-time.

Conclusions

In this contribution we present an efficient event-driven driven computation scheme based on pre-calculated characterization tables. Particularly, we describe how to embed electrical coupling in the event-driven simulation scheme. We validate the simulation approach with illustrative simulations of spiking neural networks that self-synchronize by means of inhibition and electrical coupling. We obtain results similar to those observed in neural networks simulated with time-driven schemes and realistic model described by differential equations.
This tool enables very fast simulation of large-scale systems. Therefore, it opens the door to massive simulations

that can address more specific studies on the role of chemical and electrical synaptic connections in the framework of neural population time coding and information processing in biologically realistic neural networks in real-time.

Acknowledgements

This work has been supported by the EU projects SpikeFORCE (IST-2001-35271) and SENSOPAC (IST-028056), and the Spanish National Grant (DPI-2004-07032).

References

Aho, A.V., Hopcroft, J.E., Ullman, J.D., 1974. The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading, MA.

Boucheny, C., Carrillo, R., Ros, E., Coenen, O.J.-M.D., 2005. Real-time spiking neural network: an adaptive cerebellar model. Lecture Notes Comput. Sci. 3512, 136–144.

Bower, J.M., Beeman, B., 1998. The Book of GENESIS. Springer-Verlag, New York.

Chez, C., 1991. The cerebellum. In: Kandel, E., Schwartz, J.H., Jessel (Eds.), Principles of Neural Science, 3rd ed, pp. 626– 645.

D'Angelo, E., Nieus, T., Maffei, A., Armano, S., Rossi, P., Taglietti, V., Fontana, A., Naldi, G., 2001. Theta-frequency bursting and resonance in cerebellar granule cells: experimental evidence and modelling of a slow K+-dependent mechanism. J. Neurosci. 21 (3), 759–770.

Delorme, A., Thorpe, S., 2003. SpikeNET: an event-driven simulation package for modelling large networks of spiking neurons. Netw.: Comput. Neural Syst. 14, 613–627.

Draghun, A., Traub, R.D., Schmitz, D., Jefferys, J.G.R., 1998. Electrical coupling underlies high-frequency oscillations in vitro. Nature 394, 189–192.

Furshpan, E.J., Potter, D.D., 1959. Transmission at the giant motor synapses of the crayfish. J. Physiol. 145, 289–325.

Gerstner, W., Kistler, W., 2002. Spiking Neuron Models. Single neurons. Cambridge University Press.

Gibson, J.R., Beierlein, M., Connors, B.W., 1999. Two electrically coupled inhibitory networks. Nature 402 (4), 75–79.

Hines, M.L., Carnevale, N.T., 1997. The NEURON simulation environment. Neural Comput. 9, 1179–1209.

Kepler, T.B., Marder, E., Abbott, L.F., 1990. The effect of electrical coupling on the frequency of model neuronal oscillators. Science 248, 83–85.

Koch, C., 1999. Biophysics of Computation: Information Processing in Single Neurons. Oxford University Press, New York, NY.

Kopell, N., Ermentrout, B., 2004. Chemical and electrical synapses perform complementary roles in the synchronization of interneuronal networks. Proc. Natl. Acad. Sci. U.S.A. 101 (43), 15482– 15487.

Latham, P.E., Richmond, B.J., Nelson, P.G., Nirenberg, S., 2000. Intrinsic dynamics in neuronal networks. I. Theory. J. Neurophysiol. 83 (2), 808–827.

Long, M.A., Landisman, C.E., Connors, B.W., 2004. Small clusters of electrically coupled neurons generate rhythmic in the thalamic reticular nucleus. J. Neurosci. 24 (2), 341–349.

Makino, T., 2003. A discrete-event neural network simulator for general neuron models. Neural Comput. Appl. 11, 210–223.

Mann-Metzer, P., Yarom, Y., 1999. Electrotonic coupling interacts with intrinsic properties to generate synchronized activity in cerebellar networks of inhibitory interneurons. J. Neurosci. 19 (9), 3298–3306.

Mattia, M., Guidice, P.D., 2000. Efficient event-driven simulation of large networks of spiking neurons and dynamical synapses. Neural Comput. 12, 2305–2329.

Neyton, J., Trautmann, A., 1985. Single-channel currents of an intercellular junction. Nature 317, 331–335.

Reutimann, J., Guigliano, M., Fusi, S., 2003. Event-driven simulation of spiking neurons with stochastic dynamics. Neural Comput. 15, 811–830.

Ros, E., Carrillo, R., Ortigosa, E.M., Barbour, B., Agís, R., 2006. Event-driven simulation scheme for spiking neural networks using look-up tables to characterize neuronal dynamics. Neural Comput. 18 (12) (in press).

Traub, R.D., Bibbig, A., 2000. A model of high-frequency ripples in the hippocampus based on synaptic coupling plus axon-axon gap junctions between pyramidal neurons. J. Neurosci. 20, 2086–2093.

Williams, J.W.J, 1964. Algorithm 232: Heapsort. Commun. ACM, 7, 347-348.